

COVER PAGE

Hewlett-Packard Company Docket Number:

100202598-1

Title:

SYSTEM AND METHOD OF SPECIFYING
IMAGE DOCUMENT LAYOUT DEFINITION

Inventors:

Steven J. Simske
3724 Rochdale Drive
Fort Collins, Colorado 80525

Malgorzata M. Sturgill
3743 Rochdale Drive
Fort Collins, Colorado 80525

SYSTEM AND METHOD OF SPECIFYING IMAGE DOCUMENT LAYOUT DEFINITION

TECHNICAL FIELD OF THE INVENTION

[0001] The present invention relates generally to the field of computers and computer applications, and in particular to a system and method of specifying document layout definition.

BACKGROUND OF THE INVENTION

[0002] Computers are increasingly used to handle and process documents, including documents that are composites of text, photographs, drawings, and graphic layout elements. Forms, templates, specialized scanning adapters, and specific re-purposing applications all require a very high degree of accuracy in the layout definition of these layout elements. Such very accurate layout definition of a digital document is commonly termed ground truth. The ground truth definition of a document should specify the type, location, size, resolution, and/or special treatment of these layout elements. Existing systems and methods require the user to be very hands-on in every step of the ground truth process. Further, existing systems and methods do not provide an output that is applicable to other image processing applications such as print-on-demand, document re-purposing, document classification and clustering, etc.

SUMMARY OF THE INVENTION

[0003] In accordance with an embodiment of the present invention, a method of processing an image comprises receiving a definition of at least one region in the image, where the region definition has a location specification and a type specification. The method further comprises displaying the boundaries of the at least one defined region according to its type specification, receiving a definition of a visible area in the image, the visible area definition having a specification of margins around the image, generating an image layout definition comprising the region definition and the visible area definition, and saving the image layout definition.

[0004] In accordance with yet another embodiment of the invention, a method of processing an image comprises determining a definition of at least one region in the image, the region definition having a location specification and a type specification. The method further comprises generating an image layout definition comprising the region definition, searching for an image layout definition template that best matches the generated image layout definition, and conforming the generated image layout definition to the best-matched image layout definition template.

[0005] In accordance with yet another embodiment of the invention, a system for processing an image comprises a graphical user interface operable to display the image and receive a definition of at least one region in the image, the region definition having a location specification and a type specification, the graphical user interface further operable to display the boundaries of the at least one defined region according to its type specification. The system further comprises a processor generating an image layout definition comprising the region definition.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] For a more complete understanding of the present invention, the objects and advantages thereof, reference is now made to the following descriptions taken in connection with the accompanying drawings in which:

[0007] FIGURE 1 is a simplified block diagram of an embodiment of a system and method of specifying image layout definition or ground truthing according to the present invention;

[0008] FIGURE 2 is a flowchart of an embodiment of a method of specifying image layout definition or ground truthing according to the present invention;

[0009] FIGURE 3 is a flowchart of a polygonal region definition process according to an embodiment of the present invention;

[0010] FIGURE 4 is a flowchart of a rectangular region definition process according to an embodiment of the present invention;

[0011] FIGURE 5 is a flowchart of a visible area definition process according to an embodiment of the present invention;

[0012] FIGURE 6 is a flowchart of an open image file process according to an embodiment of the present invention;

[0013] FIGURE 7 is a flowchart of a display image file process according to an embodiment of the present invention;

[0014] FIGURE 8 is a flowchart of a snap-to-template process according to an embodiment of the present invention;

[0015] FIGURE 9 is an exemplary image document that may be processed by the system and method of specifying image layout definition;

[0016] FIGURE 10 is an exemplary image document used to illustrate the functionalities of the system and method of specifying image layout definition according to an embodiment of the present invention; and

[0017] FIGURE 11 is an exemplary image document used to illustrate the functionalities of a snap-to-template process according to an embodiment of the present invention.

DETAILED DESCRIPTION OF THE DRAWINGS

[0018] The preferred embodiment of the present invention and its advantages are best understood by referring to FIGURES 1 through 11 of the drawings, like numerals being used for like and corresponding parts of the various drawings.

[0019] FIGURE 1 is a simplified block diagram of an embodiment of a system and method of specifying image layout definition 10 according to the present invention. Image ground truth is another common term used to refer to a highly accurate specification of an image document. The image document layout definition may include a number of specifications such as region segmentation, region classification, region clustering, region layout, and region modality. Region segmentation refers to the actual boundaries or outline of a region. Region classification refers to what type of data is in the region, such as text, drawing, photograph, etc. Region classification may be used for clustering, such as clustering lines of text into columns. Region layout is the specification of the relative and absolute physical location of the regions in the image document. Region modality refers to the treatment of the region as a black-and-white, gray scale, or color layout element, which also specifies the bit depth of the region. The term “layout definition” will be used to refer to a collective specification of the regions in an image document. A high degree of accuracy in the layout definition is often required in processing image documents such as re-purposing

documents, providing document templates, specialized scanning, creating document templates, and other processes.

[0020] According to an embodiment of image ground truth system 10 operating on a computer platform 12, an image document 14 is provided as input thereto. Computer platform 12 may be any device with a display and a processor. Computer platform 12 may be a portable device or a desktop device and typically comprise a pointing device such as a mouse, touch pad, touch screen, or a writing stylus. Image document 14 is preferably a scanned image of a document in a media file type such as Tag(ged) Image File Format (.TIF), Bit Map (.BMP), Graphic Interchange Format (.GIF), Portable Document Format (.PDF), Joint Photographic Experts Group (.JPEG), etc. or an electronic document in a word processing format such as WORD (.DOC), Hypertext Markup Language (HTML), or another suitable document type. Image ground truth system 10 is operable to automatically analyze document 14 and detect zones in which the document layout elements are present. The document layout elements may include text, graphics, photographs, drawings, and other visible components in the document. Alternatively, system 10 permits the user to specify, using a graphic user interface 18, the various regions occupied by these layout elements. System 10 is operable to output a specification of the image document layout definition 16 in a specified format such as eXtensible Markup Language (XML). System 10 may also output the image document layout definition as a layout template to a template database 19. Template database 19 is a repository for templates that define the layout of image documents. A template comprises a definition of the region type, modality and other properties, visible area, and other specifications of the image document. Using predefined image document templates, new image documents can be quickly put together with new text, photograph, and graphic layout elements. Furthermore, predefined templates may be used to conform image documents to correct inadvertent shifts during document scanning, for example, so that they follow a predefined format. An example of this process is shown in FIGURE 10 and described in more detail below.

[0021] Image layout definition 16 can serve as input to a variety of systems and applications. For example, image layout definition 16 may be used for document comparison and clustering/classification purposes. Further, image layout definition 16 may be used as a template for processing information. For example, image layout definition 16 may define a template with six photographic regions arranged in a certain layout. This

template may be used to arrange and layout photographs in a folder, for example. Image layout definition 16 may be easily compared with other templates or layout definition files to find the most suitable arrangement or layout of the photographs. The use of image layout definition 16 as a template also enables scanned document images that may have been slightly skewed or shifted to be corrected according to the layout specification in the template. In addition, image layout definition 16 may be used as input to a print-on-demand (POD) system that uses it to proof the layout of the documents as a measure for quality assurance. Image layout definition 16 may also be used to ensure proper rendering of a complex scanned document.

[0022] FIGURE 2 is a flowchart of an embodiment of a method 20 of specifying image layout definition or ground truthing according to the present invention. In blocks 22 and 23, a source of an image document 14 such as a stored file, a video frame, the output from a scanner, is opened and displayed, respectively. Optionally, the user may specify to resize the image file and/or to display the image file so that the entire image is shown in the available display screen. Details on these processes are described below and shown in FIGURES 6 and 7. As shown in FIGURE 9, the image file is displayed in a graphical window 23 in graphical user interface 18. Graphical user interface 18 may include a menu bar 24 comprising a plurality of menu selections 26. FIGURE 9 further shows a help pop-up window 28 that contains text describing the functionalities associated with the file menu selection. Returning to the flowchart in FIGURE 2, the user may instruct system 10 to generate region definitions by inputting the region boundaries or vertices (block 30), by a region click-and-select process (block 32), or by an automatic region analysis process (block 34).

[0023] Region click-and-select process shown in block 32 enables a user to use a pointing device to indicate on the graphical user interface the location of points within regions of interest for classification and segmentation. For example, if the user clicks on a point on the image document displayed on the graphical user interface, the region containing the identified point is analyzed and the boundaries of the region are derived. The data type of the region containing the identified point is also determined. Therefore, the user may define the regions of the image document by successively clicking on a point within each region.

[0024] Automatic region analysis process shown in block 34 is a process that performs zoning analysis on the image document to form all of its regions using a

segmentation process, and determine the region characteristics using a classification process. Various techniques are well-known in the art for performing segmentation analysis, which fall into three broad categories: top-down strategy (model-driven), bottom-up strategy (data-driven), and a hybrid of these two strategies. Examples of these strategies are described in Theo Pavlidis and Jiangying Zhou, *Page Segmentation and Classification*, published in *Document Image Analysis*, pp 226-238, 1996, and Anil K. Jain and Bin Yu, *Documentation Representation and Its Application to Page Decomposition*, published in *Pattern Analysis and Machine Intelligence*, pp 294-308, Vol. 20, No. 3, March 1998. Various techniques are well-known in the art for performing classification analysis, which are also described in the above references. Further, a suitable automatic zoning analysis process is implemented in the PrecisionScan software used in the image capture devices such as the ScanJet 5300C manufactured by Hewlett-Packard Company of Palo Alto, California.

[0025] Process 20 further provides a third method of defining the regions in the image document, as shown in block 30. The process in block 30 enables the user to define a polygonal region, a rectangular region, and a visible area in the image document. This process is described in more detail below with reference to FIGURES 3-5.

[0026] In block 36, the defined regions in document 14 are displayed in graphical user interface 18 and an example of which is shown in FIGURE 10. As shown in FIGURE 10, the boundaries of each region is outlined by color-coded lines (indicated by different lines in the figure). For example, a text region (regions 50-56) may be outlined in green, a color graphic region (regions 57) may be outlined in purple, a black and white graphic region (region 58) may be outlined in blue, a photographic region (region 59) may be outlined in yellow, etc. Further as shown in block 38, a user may provide or modify the layout definition of selected regions in the document. For example as shown in FIGURE 10, the user may select region 59, which is a region containing a photographic element. The user may do so by right-clicking on the selected region, which causes a pop-up submenu 60 to appear over region 59 that displays a number of region layout definitions that the user may modify. For example, the user may change the current region type setting of region 59 from “photo” to another region type. The user may also verify or modify the layout specification by inputting the region modality (such as black and white, gray scale or color), highlighting a specific region, and deleting a region using the same pop-up submenu. The pop-up submenu 60 is displayed in such way to indicate a current setting of the region, such as by highlighting

or deactivating the “photo” and “color” selections for the region type and modality settings, respectively. By specifying the modality of a region, the bit-depth of the region is effectively changed. For example, a black-and-white setting may equate to a 1-bit bit-depth, a gray scale setting may equate to a 8-bit bit-depth, and a color setting may equate to a 24-bit bit-depth. Therefore, by giving the user the ability to change the modality and type of each region, the same image document can be modified to be used for another purpose, which is commonly known as re-purposing.

[0027] In block 40 of FIGURE 2, if desired, the user may also update the boundaries of the defined regions by selecting the region and then drag the outline of the region boundaries to enlarge or contract the region by a process such as “rubberband boxing.” The user may also modify or specify the margins of the image document by selecting menu items associated with the visible area function, as shown in block 42. The visible area of an image document defaults to the entire image, but the user may make the visible area smaller than the entire document image. If the visible area specified by the user is too small to fully enclose any one region in the document, it is automatically expanded to include the full boundaries of all the regions in the document. A click-and-drag method can also be used to modify the visible area of the image document. The user can iteratively and selectively perform the above optional functions and save the document layout definitions, as shown in block 44. The process ends in block 46.

[0028] FIGURES 3-5 provide additional details on the processes of defining polygonal regions, rectangular regions, and visible areas in the document. FIGURE 3 is a flowchart of a polygonal region definition process 70 according to an embodiment of the present invention. Polygonal region definition process 70 provides a user the ability to generate polygonal region outlines around layout elements in the document. Generally, polygonal regions are regions with non-rectangular boundaries or regions with more complex boundaries. To create a polygonal region, the user may select a create polygon function, and then the user may indicate the vertices of the polygon around the document layout element by successive clicks of the pointing device or mouse on the displayed document in graphic user interface 18, as shown in block 72. The displayed image of the document is updated continually on the screen to provide a visual feedback of the resulting lines and vertices of the polygonal region. Process 70 may automatically close the polygonal region, in other words connect the first user-indicated vertex and the last user-indicated vertex, as shown in block

74. The user may indicate the completion of the vertices by selecting an appropriate function or by double-clicking when inputting the last vertex. The polygonal region is thus entered by the user.

[0029] At block 76, the boundaries of the generated region are verified to ensure that the enclosed region does not overlap another region in the document and that the boundary lines of the region do not cross each other, for example. A separate and independent region manager 77 may be selected to enforce the adherence to a region enforcement model. For example, one region enforcement model may specify that no regions may have overlapping boundaries, another region enforcement model may specify that a text region may be overlaid over a background region and that the text is contained completely within the background region, or another region enforcement model may specify a permissible ordering of overlapping regions and what type of layout elements those overlapping regions may contain (commonly termed “multiple z-ordering”), etc. If region irregularities have been detected, an pop-up window containing an error message is displayed. Process 70 may automatically delete the irregular region(s) or crop or shift regions so that the enforcement models are followed.

[0030] In block 78, the region type and modality and/or other definitions associated with the polygonal region are set to the default values. The default values may be determined *a priori* by the user or they may be system-wide defaults. A newly-created polygonal region may default to text and black-and-white type and modality values, respectively. These default values can be easily modified by the user to other values, such as described above and shown in FIGURE 2. A specification of the polygon region definition is generated, as shown in block 80. However, the generation of the polygonal region definition in a particular format, such as eXtensible Markup Language, may be performed when the entire document layout has been completed. The polygonal region can be saved along with the other document layout definitions of the document, as shown in block 82. The process ends in block 84.

[0031] FIGURE 4 is a flowchart of a rectangular region definition process 90 according to an embodiment of the present invention. A rectangular region is, by definition, a four-sided area with 90° corners. The user may first select a create a rectangular region function, and then indicate, using the pointing device on the graphical user interface, the first corner of the rectangle, as shown in block 92. A rubberband box is displayed on the

graphical user interface which enables the user to drag or move the opposing corner of the rectangular region in blocks 94 and 96. The boundaries of the resulting rectangular region is displayed in block 98. The boundaries of the generated rectangular region is verified by using a region manager to ensure that the resultant regions comply with a region enforcement model 101. For example, the region may not be permitted to overlap another region in the document and that the boundary lines of the region should not cross each other in block 100. Other examples of region enforcement models comprise a specification that no regions may have overlapping boundaries, a specification that a text region may be overlaid over a background region and that the text is contained completely within the background region, or a specification of permissible ordering of overlapping regions and what type of layout elements those overlapping regions may contain (commonly termed “multiple z-ordering”), etc. If region irregularities have been detected, a pop-up window containing an error message is displayed. Process 90 may automatically delete the irregular region(s) or crop or shift regions so that the enforcement models are followed. If such irregularities have been detected, a pop-up window containing an error message is displayed.

[0032] The default characteristics of the newly-created rectangular region may be set to the default values of text and black-and-white type and modality values, respectively, as shown in block 102. The newly-created rectangular region definition or the location of the rectangular region is generated and saved, along with other layout definitions of the document, as shown in block 104. The process ends in block 106.

[0033] FIGURE 5 is a flowchart of a visible area definition process 110 according to an embodiment of the present invention. As described above, the visible area definition specifies the outer boundaries around the edge of the document. In block 112, the user invokes the visible area functionality by selecting the create visible area function and indicates the first corner of the visible area. A rubberband box is then displayed in the graphical user interface to enable the user to manipulate the size (width and length) of the visible area, as shown in block 114. The user then indicates the location of the opposite corner of the visible area using the pointing device, as shown in block 116. The resulting visible area boundaries are displayed, as shown in block 118. The visible area so specified is verified, as shown in block 120. If the visible area boundaries is too small to fully enclose any one region in the document, its boundaries are automatically expanded to enclose the boundaries of all the regions in the document. The visible area definitions are generated and

saved along with other document layout element layout definitions, as shown in block 122. The process ends in block 124. The visible area layout specification is particularly important in electronic publication applications as it enables the user to specify the margins on the image, and thus the amount of white space around the boundaries of the page.

[0034] An example of a model for the layout definition specification output of process 20 is:

```
<!ELEMENT GroundTruthing (name, version, n_regions, GroundTruthingRegion+, xres,
yres, width, height)>
  <!ELEMENT name (#PCDATA)>
  <!ELEMENT version (#PCDATA)>
  <!ELEMENT n_regions (#PCDATA)>
  <!ELEMENT xres (#PCDATA)>
  <!ELEMENT yres (#PCDATA)>
  <!ELEMENT width (#PCDATA)>
  <!ELEMENT height (#PCDATA)>
  <!ELEMENT VisibleArea (left, right, top, bottom)>
    <!ELEMENT left (#PCDATA)>
    <!ELEMENT right (#PCDATA)>
    <!ELEMENT top (#PCDATA)>
    <!ELEMENT bottom (#PCDATA)>
  <!-- resolutions in ppi, width & height in pixels -->
  <!ELEMENT GroundTruthingRegion (bbox, polygon, region_type,
region_modality)>
    <!ELEMENT region_type (#PCDATA)>
    <!-- region types are TEXT | DRAWING | PHOTO | TABLE | EQUATION-->
    <!ELEMENT region_modality (#PCDATA)>
    <!-- modalities are: BW | GRAY | COLOR -->
    <!-- bbox and polygon values in pixel location for current resolutions -->
    <!ELEMENT bbox (xmin, xmax, ymin, ymax)>
      <!ELEMENT xmin (#PCDATA)>
      <!ELEMENT xmax (#PCDATA)>
      <!ELEMENT ymin (#PCDATA)>
      <!ELEMENT ymax (#PCDATA)>
    <!ELEMENT polygon (n_vertices, vertex+)>
      <!ELEMENT n_vertices (#PCDATA)>
      <!ELEMENT vertex (xcoord, ycoord)>
        <!ELEMENT xcoord (#PCDATA)>
        <!ELEMENT ycoord (#PCDATA)>
```

The above represents a model for the ground truthing metadata produced for each image document. The data is represented as integers (int), floating point (double) and enumerated types or strings. The rectangular boundaries are represented as “xmin,” “xmax,” “ymin,” and

“ymax,” and the vertices as “vertex(xcoord,ycoord).” The region type and region modality are specified by “region_type” and “region_modality.” The notation “#PCDATA” is replaced by actual data obtained by document analysis. An example of a partial layout definition specification of the exemplary image document shown in FIGURES 9 and 10 is shown below:

```
<GroundTruthing>
  <name>Ground Truthing Engine in JAVA</name>
  <version>1.0</version>
  <n_regions>1</n_regions>
  <xres>75</xres>
  <yres>75</yres>
  <width>638</width>
  <height>875</height>
  <VisibleArea>
    <left>0</left>
    <right>638</right>
    <top>0</top>
    <bottom>825</bottom>
  </VisibleArea>
  <GroundTruthingRegion>
    <region_type>PHOTO</region_type>
    <region_modality>BW</region_modality>
    <bbox>
      <xmin>329</xmin>
      <xmax>593</xmax>
      <ymin>45</ymin>
      <ymax>412</ymax>
    </bbox>
    <polygon>
      <n_vertices>5</n_vertices>
      <vertex>
        <xcoord>329</xcoord>
        <ycoord>412</ycoord>
      </vertex>
      <vertex>
        <xcoord>593</xcoord>
        <ycoord>412</ycoord>
      </vertex>
      <vertex>
        <xcoord>593</xcoord>
        <ycoord>45</ycoord>
      </vertex>
      <vertex>
        <xcoord>329</xcoord>
```

```

        <ycoord>45</ycoord>
      </vertex>
      <vertex>
        <xcoord>329</xcoord>
        <ycoord>412</ycoord>
      </vertex>
    </polygon>
  </GroundTruthingRegion>

```

...

</GroundTruthing>

It may be seen that one specified region is a region containing a photograph layout element that is to be treated as a black-and-white layout element. Its boundaries and vertices have been defined in X and Y coordinates. Further, the visible area boundaries is also defined in terms of left, right, top and bottom margins. The use of a format such as XML for the layout definition yields many advantages. Image documents may be compared with one another, classified and clustered using the layout definition specification. The layout definition specification may also be provided as input to a print-on-demand system that uses the specification to “proof” its layout and to maintain print quality.

[0035] There are instances in which a user may desire to specify the size of the image file to limit the amount of memory needed to load the image file or limit the bandwidth needed to transmit the image file over a network to a remote ground truth system. Process 140 enables the user to specify a smaller size (and thus lower resolution) of the image to use for ground truthing, as shown in FIGURE 6. In block 142, the user provides the desired image size. For example, the user may specify 1 MB as the desired file size of the image. In block 144, the bit depth or the number of bits used to scan or store data of each pixel of the image is determined. The bit depth information may be extracted from the file header of the image file, for example. The image size is then determined, as shown in block 146. The image size may be stated in terms of the two dimensions, width and height, of the document in inches for example, and may also be extracted from the file header. The X and Y resolution of the image, in response to the user-specified image size are determined, as shown in block 148, by using the following equations:

$$X_{\text{RESOLUTION}} * Y_{\text{RESOLUTION}} = \text{IMAGE_SIZE} / (\text{BIT_DEPTH} * \text{WIDTH} * \text{HEIGHT})$$

With all variables to the right of the equal-to sign known, and if the resolution is the same in the X and the Y axes, the X and Y resolution can be computed. The computed resolution may be used to open the image file as well as to transmit the image file across network links to limit the memory size or bandwidth needed to process the image file. The process ends in block 150.

[0036] FIGURE 7 is a flowchart of a display image file process 160 according to an embodiment of the invention. Process 160 may be used so that the entire image document is simultaneously displayed on the screen without requiring the user to scroll or manipulate the display to see all the pixels. In block 162, the resolution of the display screen is determined. For example, the display screen resolution may be set to 1024 pixels x 768 pixels. The number of pixels occupied by the frame, menu bar, and other layout elements of graphical user interface 18 (FIGURE 1) is determined so that the available space to display the image document is computed in block 164. For example, the window frame, tool bars, menu bars and other graphical user interface components may reduce the amount of space available to display the image to 974 x 668 pixels. The viewed image size is then determined, as shown in block 166. The image size may be stated in terms of the two dimensions, width and height, of the document in inches for example, and may also be extracted from the file header. For example, the viewed image size may be 8.5 inches by 11 inches. The maximum resolution available in the X- and Y-axes for viewing the image on the display screen is then computed, and the smaller resolution of the two is selected as the viewing resolution, as shown in block 168. The process ends in block 170.

[0037] In the 1024 x 768 display screen resolution example above, if the space available to display the image after accounting for the graphical user interface is 978 x 668 pixels, and the size of the image is 8.5 in. x 11 in., then the maximum resolution in the Y-axis is $668/11$, which equals to 60.7 pixels per inch (PPI). In the X-axis, the maximum resolution is $974/8.5$, which equals to 114.6 pixels per inch. Therefore, 60 pixels per inch is the selected resolution to display the image so that it can be viewed on the screen in its entirety. Because the region boundaries are defined in pixels that are easily scaled up or down in resolution, and that a user can choose integer divisor values of the original resolution to scale the boundaries, region boundary information is maintained without blurring.

[0038] FIGURE 8 is a flowchart of a snap-to-template process 180 according to an embodiment of the present invention. Snap-to-template process 180 may be used to

conform scanned images to a previously defined image document template in template database 19 (FIGURE 1), which is accessible by system 10. Referring to FIGURE 8, after the image document has been loaded into random memory accessible by system 10, and its layout definitions specified using process 20 shown in FIGURE 2 and described above, a search is made in the template database for the closest matching template, as shown in block 182. The search results is displayed to the user on the graphical user interface for the user's approval, as shown in block 184. If the search yields more than one match, then a list of matches is shown to enable the user to select one to be used in block 186. The selected template is then loaded in random memory accessible by system 10 and its regional definitions may be displayed in the graphical user interface, as shown in block 88. As shown in FIGURE 11, image document 14 comprises several text and other graphical layout elements 200-206. The best-match template comprises several defined regions 210-216 that specify the location, type, modality and other properties of the defined regions. It may be seen that because of scanning or other processing, the layout elements of image document 14 are shifted slightly as compared with the specifications of the best-match template. In block 190, the image document elements are "snapped" or conformed to the layout definition specified in the template, cropping, scaling and/or de-skewing the layout elements or document as appropriate. Where the relative or absolute location of the layout elements and/or visible area is offset from the region layout in the template, the location of the layout elements are revised. Where the region type or modality does not match that of the template, it is corrected. The modified image document now has layout elements that conform with the defined regions of the template and the modified image document is saved in memory in block 192. The process ends in block 194.

[0039] It may be seen that by using process 180, image documents can be standardized in format with very accurate region layout definitions that conform to the standard set forth in a template. The quality of the image documents so processed can be assured so that offset or skewed images can be detected and corrected. Furthermore, the treatment and processing of defined regions may be standardized according to the template.

[0040] Forms, templates, specialized scanning adapters, and specific re-purposing applications require a high degree of accuracy in layout definition. Embodiments of the present invention are operable to provide a highly accurate layout definition of an image document which specifies the location of layout elements in the image document, and

their respective types and modalities. The present invention is operable to accept user input of region specification as well as using automatic segmentation and classification analysis. The user may input the boundaries of the regions easily by clicking on a region or by defining the boundaries of the region using the graphical user interface. The layout definition output in eXtensible Markup Language format can be easily manipulated, processed, or used by other applications. The layout definition output may also be used as a image document template that can be used to conform subsequent image documents as to the location of the regions and visible areas as well as the region type and modality. The use of a region management models also enables a user to conform the image document regions to the selected model. Further, the present invention enables a user to process a lower-resolution version of the image document to save on memory usage, processing resources and/or transmission bandwidth.